[3. Compare conditions and booleans](https://developer.android.com/codelabs/android-development-kotlin-1.2?continue=https%3A%2F%2Fdeveloper.android.com%2Fcourses%2Fpathways%2Fandroid-development-with-kotlin-1%23codelab-https%3A%2F%2Fdeveloper.android.com%2Fcodelabs%2Fandroid-development-kotlin-1.2" \l "2)

In this task, you learn about booleans and checking conditions in the Kotlin programming language. Like other languages, Kotlin has booleans and boolean operators such as less than, equal to, greater than, and so on (<, ==, >, !=, <=, >=).

1. Write an if/else statement.

val numberOfFish = 50  
val numberOfPlants = 23  
if (numberOfFish > numberOfPlants) {  
    println("Good ratio!")   
} else {  
    println("Unhealthy ratio")  
}

⇒ Good ratio!

1. Kotlin offers the ability to easily define a succession of values with starting and terminating endpoints. This is called a range. The easiest way to create a range in Kotlin is with the ".." operator. Try using a range in an if statement. In Kotlin, the condition you test can use ranges, too.

val fish = 50  
if (fish in 1..100) {  
    println(fish)  
}

⇒ 50